

Efficient learning in ABC algorithms

Mohammed Sedki

Institut de Mathématiques et Modélisation de Montpellier
Université Montpellier 2, France

Jean-Marie Cornuet

Centre de Biologie et Gestion de Populations,
INRA, Montpellier, France

Jean-Michel Marin^{*†} and Pierre Pudlo

Institut de Mathématiques et Modélisation de Montpellier
Université Montpellier 2, France

Christian P. Robert

CEREMADE, Université Paris Dauphine, France
Institut Universitaire de France
CREST, INSEE, France

Abstract

Approximate Bayesian Computation has been successfully used in population genetics models to bypass the calculation of the likelihood. These algorithms provide an accurate estimator by comparing the observed dataset to a sample of datasets simulated from the model. Although parallelization is easily achieved, computation times for assuring a suitable approximation quality of the posterior distribution are still long. To alleviate this issue, we propose a sequential algorithm adapted from [Del Moral et al. \(2012\)](#) which runs twice as fast as traditional ABC algorithms. Its parameters are calibrated to minimize the number of simulations from the model.

Keywords: likelihood-free sampler, Sequential Monte Carlo, Approximate Bayesian Computation, population genetics

^{*}Corresponding author: place Eugène Bataillon, Case Courrier 051, 34095 Montpellier cedex 5

[†]JEAN-MICHEL.MARIN@UNIV-MONTP2.FR

1 Introduction

We work in the parametric Bayesian setting and are interested in the posterior distribution of $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^d$. Let $\pi(\boldsymbol{\theta})$ denote the prior distribution and $f(\mathbf{x}|\boldsymbol{\theta})$ the likelihood. The probability density function of the target posterior is given as

$$\pi(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}}) \propto \pi(\boldsymbol{\theta})f(\mathbf{x}_{\text{obs}}|\boldsymbol{\theta}) ,$$

where $\mathbf{x}_{\text{obs}} \in \mathcal{D}$ is the observed dataset, not necessarily an independent and identically distributed (iid) sample. For complex models, the computation of values of the likelihood function $f(\mathbf{x}_{\text{obs}}|\cdot)$ can be expensive or plain impossible which renders sampling from $\pi(\cdot|\mathbf{x}_{\text{obs}})$ extremely difficult. Approximate Bayesian Computation (ABC) is a recent technique that only requires sampling from $f(\cdot|\boldsymbol{\theta})$ (Marin et al., 2012). This methodology comes from the population genetics community. We refer the reader to Marin et al. (2012) and Beaumont (2010) which provides an excellent survey of ABC methods and its uses in biology. For more practical considerations, one can also consult Csilléry et al. (2010). The likelihood-free rejection sampler of Pritchard et al. (1999) works as follows:

```

for  $i = 1$  to  $N$  do
  repeat
    Generate  $\boldsymbol{\theta}'$  from the prior distribution  $\pi(\cdot)$ 
    Generate  $\mathbf{z}$  from  $f(\cdot|\boldsymbol{\theta}')$ 
  until  $d(S(\mathbf{z}), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon$ 
   $\boldsymbol{\theta}_i \leftarrow \boldsymbol{\theta}'$ 
end for

```

We simulate first a parameter value $\boldsymbol{\theta}'$ from the prior, then a dataset \mathbf{z} from the corresponding model. We subsequently compare the simulated dataset to the observed one \mathbf{x}_{obs} via some summary statistics $S : \mathcal{D} \rightarrow \mathcal{S}$ and a distance function $d : \mathcal{S} \times \mathcal{S} \rightarrow \mathbb{R}^+$. If \mathbf{z} is close enough to \mathbf{x}_{obs} according to a tolerance level ε , we keep $\boldsymbol{\theta}'$. The likelihood-free rejection algorithm samples from

$$\pi_\varepsilon(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}}) \propto \int_{\left\{ \mathbf{z} \in \mathcal{D} \mid d(S(\mathbf{z}), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon \right\}} \pi(\boldsymbol{\theta})f(\mathbf{z}|\boldsymbol{\theta})d\mathbf{z} , \quad (1)$$

the marginal distribution of $\pi_\varepsilon(\boldsymbol{\theta}, \mathbf{z}|\mathbf{x}_{\text{obs}})$. The idea behind ABC is that the summary statistics coupled with a small tolerance should provide a good approximation of the posterior distribution $\pi_\varepsilon(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}}) \approx \pi(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}})$. Marjoram et al. (2003) have introduced a Monte Carlo Markov Chain (MCMC) algorithm targeted at (1) which does not require any calculation of the likelihood.

Rejection sampling and MCMC methods can perform poorly if the tolerance level ε is small. In practice, the tolerance level ε used in the rejection sampling algorithm is not fixed in advance, but corresponds to a quantile of the distances between the observed dataset and some simulated ones. Various sequential Monte Carlo algorithms (Doucet et al., 2001; Del Moral, 2004; Liu, 2008) have been constructed as an alternative to these two methods, see Sisson et al. (2007), Sisson et al. (2009), Beaumont et al. (2009), Drovandi and Pettitt (2011) and Del Moral et al. (2012). These algorithms start from a large tolerance level ε_0 , and at each iteration the tolerance levels decreases, $\varepsilon_t < \varepsilon_{t-1}$. The simulation problem becomes therefore more and more difficult, whereas the proposal distribution for the parameters $\boldsymbol{\theta}$ becomes more and more accurate.

The algorithm of Beaumont et al. (2009) corrects the bias of the one of Sisson et al. (2007) (Sisson et al., 2009) and is a particular Population Monte Carlo scheme (Cappé et al., 2004). It requires fixing a sequence of decreasing tolerance levels $\varepsilon_0 > \varepsilon_1 > \dots > \varepsilon_T$ which is not very realistic for practical problems. In contrast, the proposals of Del Moral et al. (2012) and Drovandi and Pettitt (2011) are adapted likelihood-free versions of the Sequential Monte Carlo sampler (Del Moral et al., 2006) and include a self-calibration mechanism for the sequence of decreasing tolerance levels.

We consider here the specific case where all the others calculations are negligible compared to a random generation from the model. This is typically true for complex models, e.g. for complex scenarios in population genetics. We adapt the likelihood-free SMC sampler of Del Moral et al. (2012) such that the number of simulated values from the model is as small as possible.

We present a naive likelihood-free sequential algorithm and we recall the proposals of Del Moral et al. (2012) and Drovandi and Pettitt (2011) in Section 2. Then we introduce our self-calibrated strategy in Section 3. Finally, we illustrate its numerical behavior on simulated data and a challenging real-data example from population genetics.

2 Sequential likelihood-free schemes

As already explained, the likelihood-free rejection scheme consists in generating a sample $\{(\boldsymbol{\theta}_i, \mathbf{x}_i), 1 \leq i \leq N\}$ from $\pi(\boldsymbol{\theta})f(\mathbf{x}|\boldsymbol{\theta})$ and in keeping the $\boldsymbol{\theta}_i$'s such that $d(S(\mathbf{x}_i), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon$. We obtain the collection $\{\boldsymbol{\theta}_j\}_{j \in J(\varepsilon, N)}$, where $J(\varepsilon, N) = \{j \in \{1, \dots, N\} : \mathbf{x}_j \in \{\mathbf{z} \in \mathcal{D} | d(S(\mathbf{z}), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon\}\}$. If ε is too small, the set $J(\varepsilon, N)$ is almost empty. On the contrary, if ε is too large then $\pi_\varepsilon(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}})$ can be very far from the posterior distribution. In practice, a quantile α of the distances $d(S(\mathbf{x}_i), S(\mathbf{x}_{\text{obs}}))$ is

chosen which corresponds to setting ε such that $J(\varepsilon, N)$ contains $\lfloor \alpha N \rfloor$ distinct indices (where $\lfloor a \rfloor$ denotes the integer part of a). With such a strategy, the θ_i 's are always sampled from the prior distribution. This can be very inefficient, especially when $\pi(\theta)$ is not informative. Sequential schemes aim at constructing proposals for the parameters which iteratively gain relevance. We start by introducing a naive approach based on the SMC sampler of [Del Moral et al. \(2006\)](#).

From now on, we assume a prior $\pi(\theta)$ that is uniform over the compact set Θ_{prior} . In an initial step, all sequential strategies employ the likelihood-free rejection sampler to create a sample from the distribution $\pi_{\varepsilon_0}(\theta|\mathbf{x}_{\text{obs}})$. In the further steps, a population of N particles $\mathcal{P}^t := \{(\theta_i^t, \mathbf{x}_i^t), 1 \leq i \leq N\}$ drawn from the distribution $\pi_{\varepsilon_t}(\theta|\mathbf{x}_{\text{obs}})$ is given at iteration $t + 1$. The naive strategy at iteration $t + 1$ is based on a fixed sequence of quantiles $\{\alpha_t\}_{t \geq 0}$ (α_0 is associated to ε_0) and a given target value for ε :

1. Sort the particles $\{(\theta_i^t, \mathbf{x}_i^t), 1 \leq i \leq N\}$ with respect to their distances from \mathbf{x}_{obs} , i.e.
$$d(S(\mathbf{x}_1^t), S(\mathbf{x}_{\text{obs}})) \leq d(S(\mathbf{x}_2^t), S(\mathbf{x}_{\text{obs}})) \leq \dots \leq d(S(\mathbf{x}_N^t), S(\mathbf{x}_{\text{obs}}))$$
2. Calibrate ε_{t+1} :
Set ε_{t+1} such that $\varepsilon_{t+1} = d(S(\mathbf{x}_{\lfloor \alpha_{t+1} N \rfloor}^t), S(\mathbf{x}_{\text{obs}}))$
3. Compute the new particle system:
 - (a) Create a new set of N particles \mathcal{P}^* in the following way:
 - Repeat $\lfloor 1/\alpha_{t+1} \rfloor$ times the first $\lfloor \alpha_{t+1} N \rfloor$ particles of \mathcal{P}^t
 - Draw the $N - \lfloor 1/\alpha_{t+1} \rfloor \lfloor \alpha_{t+1} N \rfloor$ last particles of \mathcal{P}^* randomly among $\{(\theta_i^t, \mathbf{x}_i^t), 1 \leq i \leq \lfloor \alpha_{t+1} N \rfloor\}$
 - (b) Move each particle $(\theta_i^*, \mathbf{x}_i^*)$ of \mathcal{P}^* according to a MCMC kernel K_t with stationary distribution $\pi_{\varepsilon_{t+1}}(\theta, \mathbf{z}|\mathbf{x}_{\text{obs}})$ to get a new set of particles \mathcal{P}^{t+1}
4. If $\varepsilon_{t+1} < \varepsilon$, return \mathcal{P}^{t+1} and ε_{t+1} as output of the algorithm. Otherwise, set $t = t + 1$ and return to step 1.

Moving according to K_t .

We propose to use a Metropolis-Hastings kernel in step 4. For instance, we can compute the empirical variance τ_t^2 of the $\{\theta_i^t\}$'s in the set \mathcal{P}_t . For each particle $(\theta_i^*, \mathbf{x}_i^*)$ of \mathcal{P}^* , we generate some $\tilde{\theta}$ according to the Gaussian distribution $\mathcal{N}(\theta_i^*, 2\tau_t^2)$ and some $\tilde{\mathbf{x}}$ according to the likelihood $f(\mathbf{x}|\tilde{\theta})$. Since the prior is uniform, the acceptance rate for such

a Gaussian proposal is binary, see [Del Moral et al. \(2012\)](#) for more details. If $\tilde{\boldsymbol{\theta}} \in \Theta_{\text{prior}}$ and $d\left(S(\tilde{\mathbf{x}}), S(\mathbf{x}_{\text{obs}})\right) \leq \varepsilon_{t+1}$, we set $(\boldsymbol{\theta}_i^{t+1}, \mathbf{x}_i^{t+1}) = (\tilde{\boldsymbol{\theta}}, \tilde{\mathbf{x}})$. Otherwise, we do not move and set $(\boldsymbol{\theta}_i^{t+1}, \mathbf{x}_i^{t+1}) = (\boldsymbol{\theta}_i^*, \mathbf{x}_i^*)$.

No Rao-Blackwellization.

We might replace Step 3.(a) by the following alternative: Given \mathcal{P}^t , the system \mathcal{P}^* consists of N iid particles generated from a uniform distribution over the $\lfloor \alpha_t N \rfloor$ first particles of \mathcal{P}^t . If $N - \lfloor \lfloor 1/\alpha_t \rfloor \alpha_t N \rfloor = 0$, Step 3.(a) corresponds to a conditional expectation under this alternative: we integrate over the uniform draws from the $\lfloor \alpha_t N \rfloor$ first particles of \mathcal{P}^t .

Quality of the output.

A sequential schemes returns weighted particles $(\mathbf{x}_1, \omega_1), \dots, (\mathbf{x}_N, \omega_N)$ associated to a target distribution Q such that

$$\sum_{i=1}^N \omega_i \delta_{\mathbf{x}_i}(A) \rightarrow Q(A),$$

for any Q -measurable set A . One way to measure the quality of a weighted sample is the Effective Sample Size (ESS) criterion, a quality indicator inspired from importance sampling ideas (cf. [\(Liu, 2008, chapter 2\)](#)). It is given by

$$\text{ESS}\left((\mathbf{x}_1, \omega_1), \dots, (\mathbf{x}_N, \omega_N)\right) = \left(\sum_{i=1}^N \omega_i^2\right)^{-1}. \quad (2)$$

In many situations, the particle system is obtained by a resampling step followed by a systematic assignment of the weight $1/N$ to each particle. Such a system can be composed of a small number of distinct particles repeated many times. Then we can not directly apply the formula (2). Indeed, ESS would equal N . In this case, we calculate the ESS by considering only the distinct particles with their aggregated weight: if one particle \mathbf{x}^* is repeated r times with the weight $1/N$, its weight after aggregation is r/N . If there are repetitions, another indicator of the quality of the weighted sample is the number of distinct particles. If the particles are repeated the same number of times, it is equal to the ESS. This indicator is an upper bound for the ESS. The difference between these two measures is even greater when the entropy of the weights is strong. If no movement is accepted at the step 3 of the above-described naive strategy, the final particle system consists of $\alpha_0 \alpha_1 \dots \alpha_T N$ distinct particles with the same weight. Each one is repeated many times in the final system. In other words, this is equivalent to running the likelihood-free rejection sampler with a quantile α equal to $\alpha_0 \alpha_1 \dots \alpha_T$. We thus obtain an ESS that

corresponds to the number of distinct particles,

$$\text{ESS}_T = \alpha_0 \alpha_1 \dots \alpha_T N,$$

which can indicate very poor quality. If now we assume that at the iteration t a proportion ρ_t of particles is accepted in the step 3 (b) of the naive strategy, the number of distinct particles is

$$\alpha_0(\alpha_1 + \rho_1)(\alpha_2 + \rho_2) \dots (\alpha_T + \rho_T)N. \quad (3)$$

To explain this value of ESS, we assume that at iteration t the system \mathcal{P}^t contains N_t distinct particles and that $\alpha_t = 1/\ell_t$ with an integer divisor ℓ_t of N_t . If each particle $(\tilde{\theta}, \tilde{\mathbf{x}})$ proposed during step 3 is accepted in the system \mathcal{P}^{t+1} with probability ρ_t , then the system \mathcal{P}^* consists of ℓ_t copies of the $\lfloor \alpha_t N \rfloor$ first particles of \mathcal{P}^t . Hence the mean number of distinct particles in the system \mathcal{P}^{t+1} is

$$\alpha_t N_t + \sum_{i=1}^{\ell_t} \alpha_t \rho_t N_t = \alpha_t N_t + \ell_t \alpha_t \rho_t N_t = (\alpha_t + \rho_t) N_t.$$

The likelihood-free SMC proposals of [Del Moral et al. \(2012\)](#) and [Drovandi and Pettitt \(2011\)](#).

The proposal of [Del Moral et al. \(2012\)](#) is an adaptive scheme that calibrates the threshold ε_t by controlling the ESS during the iterations. The threshold ε_t is chosen such that

$$\text{ESS}_{t+1} = \alpha \text{ESS}_t,$$

where $\alpha \in]0, 1[$. In practice, they consider the number of particles in the iteration t for which the distance to the observation is lower than ε_{t+1} . This quantity is treated as the ESS and repetitions are not taken into account. The parameter α is a quality index. If $\alpha \approx 1$ then the scheme will move slowly towards the target ABC and the successive approximations are good. Conversely, if α is close to 0, the stabilization at the desired level is very fast but the quality of the successive approximations is poor. In their numerical experiments, they keep α between 0.9 and 0.99. This scheme is not optimized to limit the number of simulations according to the model. Typically, for the same parameter value they propose to draw M samples from the augmented ABC target

$$\pi_\varepsilon(\mathbf{x}_{1:M}, \boldsymbol{\theta}) \propto \pi(\boldsymbol{\theta}) \prod_{k=1}^M f(\boldsymbol{\theta} | \mathbf{x}_k) \frac{1}{M} \sum_{k=1}^M \mathbf{1} \left\{ d(S(\mathbf{x}_k), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon \right\}.$$

Finally, we note that they perform a resampling step before the kernel shifting step only if the ESS falls below a certain threshold.

The proposal of [Drovandi and Pettitt \(2011\)](#) applies the naive strategy by fixing the sequence of quantiles α_t to a unique value $1 - \alpha$. Only the $\lfloor \alpha N \rfloor$ particles with the largest distance from the observation are modified. This scheme is also clearly not calibrated to minimize the number of simulations according to the model.

3 Our self-calibrated algorithm

The main idea of our method is to choose the minimal ε_{t+1} for which $(\alpha_t + \rho_t) \geq 1$. In the above algorithm, we then have to permute step 3 (in which ε_{t+1} is fixed) and step 4 (in which we can compute ρ_t). Unfortunately the MCMC kernel K_t of step 4 depends heavily on ε_{t+1} , hence this permutation is not trivial. Similar to the naive approach, we assume that at iteration $t+1$ we are given a population of N particles $\mathcal{P}^t := \{(\boldsymbol{\theta}_i^t, \mathbf{x}_i^t), 1 \leq i \leq N\}$ drawn from the distribution $\pi_{\varepsilon_t}(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}})$. At iteration $t+1$, our self-calibrated likelihood-free SMC sampler works as follows:

1. Sort the particles $\{(\boldsymbol{\theta}_i^t, \mathbf{x}_i^t), 1 \leq i \leq N\}$ with respect to their distances to \mathbf{x}_{obs} , i.e.

$$d_1 = d(S(\mathbf{x}_1^t), S(\mathbf{x}_{\text{obs}})) \leq d_2 = d(S(\mathbf{x}_2^t), S(\mathbf{x}_{\text{obs}})) \leq \dots \leq d_N = d(S(\mathbf{x}_N^t), S(\mathbf{x}_{\text{obs}}))$$

2. Calibration of ε_{t+1} :

Compute σ_t^2 , the variance of $\{\boldsymbol{\theta}_i^{t-1}, 1 \leq i \leq N\}$ in the particle system \mathcal{P}^{t-1} .

$\alpha' \leftarrow 0.0$

repeat

$\alpha' \leftarrow \alpha' + 0.01$

$\varepsilon(\alpha') \leftarrow d_{\lfloor \alpha' N \rfloor}$

for $i = 1$ **to** $\lfloor \alpha' N \rfloor$ **do**

if particle $(\tilde{\boldsymbol{\theta}}_i, \tilde{\mathbf{x}}_i)$ does not exist **then**

Generate $\tilde{\boldsymbol{\theta}}_i$ from the normal distribution $\mathcal{N}(\boldsymbol{\theta}_i^t, 2\sigma_t^2)$ and $\tilde{\mathbf{x}}_i$ from $f(\mathbf{x}|\tilde{\boldsymbol{\theta}}_i)$

end if

end for

Set $N'(\alpha')$ as the number of particles $(\tilde{\boldsymbol{\theta}}_i, \tilde{\mathbf{x}}_i)$ among the $\lfloor \alpha' N \rfloor$ first ones that satisfy $\tilde{\boldsymbol{\theta}}_i \in \Theta_{\text{prior}}$ and $d(S(\tilde{\mathbf{x}}_i), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon'(\alpha')$

$\rho'(\alpha') \leftarrow N'(\alpha') / \lfloor \alpha' N \rfloor$

until $\alpha' + \rho'(\alpha') \geq 0.9$

$\alpha_{t+1} \leftarrow \alpha'$, $\varepsilon_{t+1} \leftarrow \varepsilon'(\alpha')$ and $\rho_{t+1} \leftarrow \rho'(\alpha')$

3. Computation of the new particle system:

```

for  $i = 0$  to  $\lfloor \alpha_{t+1}N \rfloor$  do
  if  $\tilde{\theta}_i \in \Theta_{\text{prior}}$  and  $d\left(S(\tilde{\mathbf{x}}_i), S(\mathbf{x}_{\text{obs}})\right) \leq \varepsilon_{t+1}$  then
     $(\theta_i^{t+1}, \mathbf{x}_i^{t+1}) \leftarrow (\tilde{\theta}_i, \tilde{\mathbf{x}}_i)$ 
  else
     $(\theta_i^{t+1}, \mathbf{x}_i^{t+1}) \leftarrow (\theta_i^t, \mathbf{x}_i^t)$ 
  end if
end for
for  $i = \lfloor \alpha_{t+1}N \rfloor + 1$  to  $N$  do
  Generate an integer number  $I$  uniformly between 1 and  $\lfloor \alpha_{t+1}N \rfloor$ 
  Generate  $\tilde{\theta}$  from the normal distribution  $\mathcal{N}(\theta_I^t, 2\sigma_t^2)$  and  $\tilde{\mathbf{x}}$  from  $f(\mathbf{x}|\tilde{\theta})$ 
  if  $\tilde{\theta} \in \Theta_{\text{prior}}$  and  $d\left(S(\tilde{\mathbf{x}}), S(\mathbf{x}_{\text{obs}})\right) \leq \varepsilon_{t+1}$  then
     $(\theta_i^{t+1}, \mathbf{x}_i^{t+1}) \leftarrow (\tilde{\theta}, \tilde{\mathbf{x}})$ 
  else
     $(\theta_i^{t+1}, \mathbf{x}_i^{t+1}) \leftarrow (\theta_I^t, \mathbf{x}_I^t)$ 
  end if
end for

```

4. If $\rho_{t+1} \leq 0.1$, return \mathcal{P}^{t+1} and ε_{t+1} as output of the algorithm. Otherwise, set $t = t + 1$ and return to step 1.

Adaptive scheme and final tolerance level.

Clearly, $\rho'(\alpha')$ increases with α' since the conditions $\tilde{\theta}_i \in \Theta$ and $d\left(S(\tilde{\mathbf{x}}_i), S(\mathbf{x}_{\text{obs}})\right) \leq \varepsilon'(\alpha')$ become less restrictive and the Metropolis-Hastings kernel moves a larger proportion of particles. Hence $\alpha' + \rho'(\alpha')$ increases with α' .

Two important questions need closer examination:

1. What is the behavior of the threshold of acceptance ε at each iteration when N tends to infinity?
2. What can we say about the final level of acceptance in our scheme?

When N increases, the calibrated tolerance level ε_t does not converge to 0, but to a positive quantity that depends only on the prior π , the likelihood f and the transition mechanisms K_t during the Metropolis-Hastings step. At each stage of the iterative algorithm, the triplet $(\alpha, \rho, \varepsilon)$ takes values on a one-dimensional manifold. If we parametrize this manifold by ε , the calibration scheme chooses the tolerance level ε such that $\alpha(\varepsilon) + \rho(\varepsilon) = 1$.

Consider for instance the $(t + 1)$ th iterative stage of the algorithm. When $N \rightarrow \infty$, the empirical distribution of the particle system \mathcal{P}^t converges to the joint distribution $\pi_{\varepsilon_t}(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}})f(\mathbf{x}|\boldsymbol{\theta})$. Hence asymptotically $\alpha(\varepsilon)$ is the cumulative distribution function of $d(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}}))$ when $(\boldsymbol{\theta}, \mathbf{x})$ is drawn from $\pi_{\varepsilon_t}(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}})f(\mathbf{x}|\boldsymbol{\theta})$. In other words,

$$\alpha(\varepsilon) = \iint \pi_{\varepsilon_t}(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}})f(\mathbf{x}|\boldsymbol{\theta})\mathbf{1}_{\left\{d\left(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}})\right)\leq\varepsilon\right\}}d\mathbf{x}d\boldsymbol{\theta}, \quad \text{for any } \varepsilon > 0.$$

The rejection step of the $(t + 1)$ th iterative stage keeps a proportion $\alpha(\varepsilon)$ of the $\boldsymbol{\theta}$'s. Hence, the empirical distribution of those $\boldsymbol{\theta}$'s converges to

$$\pi_{\varepsilon}(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}}) = \alpha(\varepsilon)^{-1} \int \pi_{\varepsilon_t}(\boldsymbol{\theta})f(\mathbf{x}|\boldsymbol{\theta})\mathbf{1}_{\left\{d\left(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}})\right)\leq\varepsilon\right\}}d\mathbf{x}, \quad \text{for any } \varepsilon > 0.$$

Asymptotically, the acceptance rate $\rho(\varepsilon)$ for the Metropolis-Hastings step associated to the Markov kernel K_{t+1} is

$$\rho(\varepsilon) = \iiint \pi_{\varepsilon}(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}})K_{t+1}(\boldsymbol{\theta}^*|\boldsymbol{\theta})f(\mathbf{x}^*|\boldsymbol{\theta}^*)\mathbf{1}_{\left\{d\left(S(\mathbf{x}^*), S(\mathbf{x}_{\text{obs}})\right)\leq\varepsilon\right\}}d\mathbf{x}^*d\boldsymbol{\theta}^*d\boldsymbol{\theta}.$$

Therefore the calibrated tolerance level ε_{t+1} is asymptotically the solution of

$$\begin{aligned} & \iint \pi_{\varepsilon_t}(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}})f(\mathbf{x}|\boldsymbol{\theta})\mathbf{1}_{\left\{d\left(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}})\right)\leq\varepsilon\right\}}d\mathbf{x}d\boldsymbol{\theta} \\ & + \iiint \pi_{\varepsilon}(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}})K_{t+1}(\boldsymbol{\theta}^*|\boldsymbol{\theta})f(\mathbf{x}^*|\boldsymbol{\theta}^*)\mathbf{1}_{\left\{d\left(S(\mathbf{x}^*), S(\mathbf{x}_{\text{obs}})\right)\leq\varepsilon\right\}}d\mathbf{x}^*d\boldsymbol{\theta}^*d\boldsymbol{\theta} = 1. \end{aligned}$$

In turn, calibrated parameters of the other iterative stages are consistent estimators of positive quantities that depend only on the prior, likelihood and the kernel inside Metropolis-Hastings. A full and formal proof should show the convergence of these empirical quantities. Calibrated parameters converge when one reaches the asymptotic regime with large N (see numerical experiments on figures 1 and 2). The moment of reaching the stopping criterion ($\rho \leq 0.1$) does not vary significantly across independent runs of our scheme. This convergence is illustrated on the toy example of [Sisson et al. \(2007\)](#) in Section 4.1.

For the analysis of the quality of the final threshold ε when the number of particles N increases to infinity, we assume that the final level of acceptance of our ABC-SMC scheme does not vary significantly from one instance to another. The ABC approximation given by the output level ε_T (where T is the stopping time) may not be low enough to assure a good approximation of the posterior. Actually, ε is often too large and we have to add an extra rejection step on the output of the sequential algorithm to decrease ε . Adding an extra rejection step is faster than violating the stopping criterion and increasing the number of iterations. In addition, the ESS of the resulting simulated sample of parameter-dataset pairs depends linearly on N (before and after the extra rejection step).

Remember that the ABC approximation π_ε is reconstructed from the final sample via classical methods (kernel density estimate, local linear regression, ...). We can infer its characteristics such as the mean, the median or quantiles with classical sample-based estimators. The reconstruction error depends heavily on the ESS of the simulated sample. Increasing N increases the quality and accuracy of the reconstruction. When a good ABC approximation is desired, we can distribute the computational cost by running the instances of our ABC-SMC scheme in parallel and merging the outputs. The level of acceptance of the resulting sample is determined as the greatest level among the distributed samples.

High Performance Computing.

A major advantage of the classical ABC algorithm is its aptitude for parallelization on multicore architectures and computing clusters. In our algorithm, we could conduct the simulations according to the probabilistic model in parallel. However this demands for a shared memory architecture to run the sequential algorithm since each iteration depends on the previously simulated sample. If the simulation of a dataset from the model and the computation of the summary statistics are slow, we observe a good parallel program performance on a multicore computer using the OpenMP API (see <http://openmp.org>): The algorithm runs almost eight times faster when using eight cores than when using a single core.

The situation is different if one wants to parallelize the program on several nodes of a cluster. The overhead due to communication and synchronization between the nodes might be substantial, with the trickiest part being the calibration of the tolerance level at each step. In this case, we recommend independent runs of the algorithm on each node of the cluster, providing independent outputs we can easily merge. In particular, we assume that we are able to construct random number generators on each node of the cluster which are mutually independent.

Efficient learning.

The aim of iterative algorithms is to learn progressively how to draw $\boldsymbol{\theta}$ without introducing bias into the computation of the posterior. Indeed, simulating $\boldsymbol{\theta}$ from the prior distribution can lead to simulated data sets far from the observation. In our proposal, we learn the current ABC approximation of the posterior from the previously simulated pairs $(\boldsymbol{\theta}, \mathbf{x})$ and we draw pairs $(\boldsymbol{\theta}, \mathbf{x})$ from this current approximation in order to gradually decrease the tolerance level ε . Clearly, applying the learning scheme is inefficient if the posterior distribution is almost equal to the prior distribution (see the discussion on the toy example of [Sisson et al. \(2007\)](#) in section 4.1). By the way, we can interpret the Metropolis-Hastings step as drawing $\boldsymbol{\theta}'$ from a kernel density estimate of the current ABC approximation of the posterior. Actually, we draw the parameter $\boldsymbol{\theta}'$ of the proposed couple $(\boldsymbol{\theta}', \mathbf{x}')$ from the Gaussian mixture

$$\frac{1}{\lfloor \alpha_t N \rfloor} \sum_{i=1}^{\lfloor \alpha_t N \rfloor} \varphi\left(b_t^{-1}(\boldsymbol{\theta}' - \boldsymbol{\theta}_i^{t-1})\right),$$

where $b_t = \sqrt{2\sigma_t^2}$ is the smoothing parameter. This mixture is a smooth density estimate of $\{\boldsymbol{\theta}_i^{t-1}, 1 \leq i \leq \lfloor \alpha_t N \rfloor\}$ whose empirical distribution is a proxy for π_{ε_t} .

Comparison with the classical ABC algorithm. Our ABC-SMC scheme returns a final particles system $\{(\boldsymbol{\theta}_i^T, \mathbf{x}_i^T), 1 \leq i \leq N\}$ with ESS_T and final acceptance level ε_T . We consider this final output as an iid simulated sample of size ESS_T . The time complexity of our algorithm in the setup of population genetics is determined by the total number of datasets simulated from the model. We compare the tolerance level reached by our algorithm with the one reached by the classical acceptance-rejection method while demanding the same computational effort and the same quality of the output in terms of ESS. We denote by N_T the total number of simulations according the model required for our scheme. The classical ABC algorithm with the same time complexity draws N_T pairs $(\boldsymbol{\theta}, \mathbf{x})$ from $\pi(\boldsymbol{\theta})f(\mathbf{x}|\boldsymbol{\theta})$. The output of this classical algorithm should be of size ESS_T to be comparable to the output of our proposal. Hence one should compare ε_T with the quantile $\varepsilon_{\text{acc-rej}}(ESS_T/N_T)$ of order ESS_T/N_T of $D = d(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}}))$ when $(\boldsymbol{\theta}, \mathbf{x})$ is drawn from the joint $\pi(\boldsymbol{\theta})f(\mathbf{x}|\boldsymbol{\theta})$. To quantify the efficiency of our scheme compared to the acceptance-rejection algorithm, we compute the gain factor. We compute the number of particles $N_{\text{acc-rej}}$ needed when using the ABC acceptance-rejection sampler to provide an approximation with level ε_T and ESS_T particles accepted. The gain factor of our ABC-SMC scheme in terms of the number of simulations according to the model is given by

$$r_T = \frac{N_{\text{acc-rej}}}{N_T}. \quad (4)$$

In terms of simulation according to the model, our scheme outperforms the standard acceptance-reject algorithm when this gain factor is greater than 1.

Convergence properties.

If the decreasing sequence of tolerance levels $\{\varepsilon_t\}_{t \geq 0}$ is fixed, the successive target distributions $\{\pi_{\varepsilon_t}\}_{t \geq 0}$ are not random. If moreover the Markov kernels $\{K_t\}_{t \geq 0}$ are deterministic, the convergence results on the SMC algorithm of [Del Moral et al. \(2006\)](#) can be applied directly.

If we consider the case where the Markov kernels depend on the past particles, we can provide convergence results using the recent works of [Del Moral et al. \(2011\)](#). We can also use theorems on triangular arrays of random variables given in [Douc and Moulines \(2008\)](#). Therefore, denote by $\{\xi_{N,i}^t = (\theta_{N,i}^t, \mathbf{x}_{N,i}^t), 1 \leq i \leq N\}$ the system of N particles at the end of the t -th iteration of the algorithm. If we introduce the sub- σ -field $\mathcal{F}_N^{t-1} = \sigma(\xi_{N,i}^{t-1}, 1 \leq i \leq N)$, the particles $\{\xi_{N,i}^t, 1 \leq i \leq N\}$ are independent given \mathcal{F}_N^{t-1} . Under standard regularity conditions, we can prove the following law of large numbers and CLT:

- If $\int |\varphi(\xi)| \pi_{\varepsilon_t}(\mathrm{d}\xi) < +\infty$, the convergence in probability

$$\frac{1}{N} \sum_{i=1}^N \varphi(\xi_{N,i}^t) \longrightarrow \pi_{\varepsilon_t}(\varphi),$$

holds with $\pi_{\varepsilon_t}(\varphi) := \int \varphi(\xi) \pi_{\varepsilon_t}(\mathrm{d}\xi)$ as N tends to infinity.

- If $\int \varphi^2(\xi) \pi_{\varepsilon_t}(\mathrm{d}\xi) < \infty$, the convergence in distribution

$$\frac{1}{\sqrt{N}} \left(\sum_{i=1}^N \varphi(\xi_{N,i}^t) - N \pi_{\varepsilon_t}(\varphi) \right) \longrightarrow \mathcal{N}(0, \delta_t^2(\varphi))$$

holds with $\delta_t^2(\varphi) := \int \varphi^2(\xi) \pi_{\varepsilon_t}(\mathrm{d}\xi) - \pi_{\varepsilon_t}(\varphi)^2$ as N tends to infinity.

4 Numerical experiments

4.1 A toy example

The following toy example, first introduced in [Sisson et al. \(2007\)](#) and widely used in the literature, consists of a simple mixture of two Gaussian distributions with different

variances and unknown mean:

$$\begin{cases} \boldsymbol{\theta} \sim \mathcal{U}_{[-10,10]}, \\ f(\mathbf{x}|\boldsymbol{\theta}) = \frac{1}{2}\phi(\mathbf{x}; \boldsymbol{\theta}, 1) + \frac{1}{2}\phi(\mathbf{x}; \boldsymbol{\theta}, 1/100), \end{cases} \quad (5)$$

where $\mathcal{U}_{[a,b]}$ denotes the uniform distribution on the interval $[a, b]$ ($a < b$) and $\phi(\cdot; \mu, \sigma^2)$ the probability density function of the one-dimensional Gaussian distribution with mean μ and variance σ^2 . As explained in [Beaumont et al. \(2009\)](#) and [Del Moral et al. \(2012\)](#), the posterior distribution associated to the observation $\mathbf{x}_{\text{obs}} = 0$ is such that

$$\pi(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}}) \propto \left\{ \phi(\boldsymbol{\theta}; 0, 1) + \phi(\boldsymbol{\theta}; 0, 1/100) \right\} \mathbf{1}_{\left\{ -10 \leq \boldsymbol{\theta} \leq 10 \right\}}. \quad (6)$$

This example is simple enough to compare the iterative methodologies.

Table 1 provides a comparison of our ABC-SMC scheme to the ones proposed by [Del Moral et al. \(2012\)](#) and [Drovandi and Pettitt \(2011\)](#) in terms of the total number of simulations according to the model necessary to provide an approximation with the same acceptance level $\varepsilon = 0.09$. The schemes proposed in [Del Moral et al. \(2012\)](#) and [Drovandi and Pettitt \(2011\)](#) are not designed to optimize the number of simulations according to the likelihood. Clearly, our algorithm performs best.

Table 1: Comparison of computational cost of the different ABC-SMC schemes.

	Cost	ESS	Final acceptance level ε
Del Moral et al. (2012)	46×10^5	29250	0.09
Drovandi and Pettitt (2011)	109×10^5	32000	0.09
Our ABC-SMC scheme	23×10^5	33285	0.09

We have applied our proposal with 10^5 particles at each iteration. Figures 1, 2 correspond to the 1st and 8th iterative stage and illustrate the convergence of the calibrated tolerance levels ε when the simulated sample size N increases. We also observed that the variance of ε is proportional to $1/N$.

We mentioned earlier that sequential schemes are not recommended when the prior provides sufficient information about the parameter. Figure 3 (two top curves) shows that when the prior density support is close around 0, the stopping criterion is met in the early iterations and the gain factor is not favorable to the use of sequential schemes. The four curves in this figure show the evolution of the gain factor introduced in (4) over the iterations. The vertical dotted line (blue) indicates the stop time. In each of

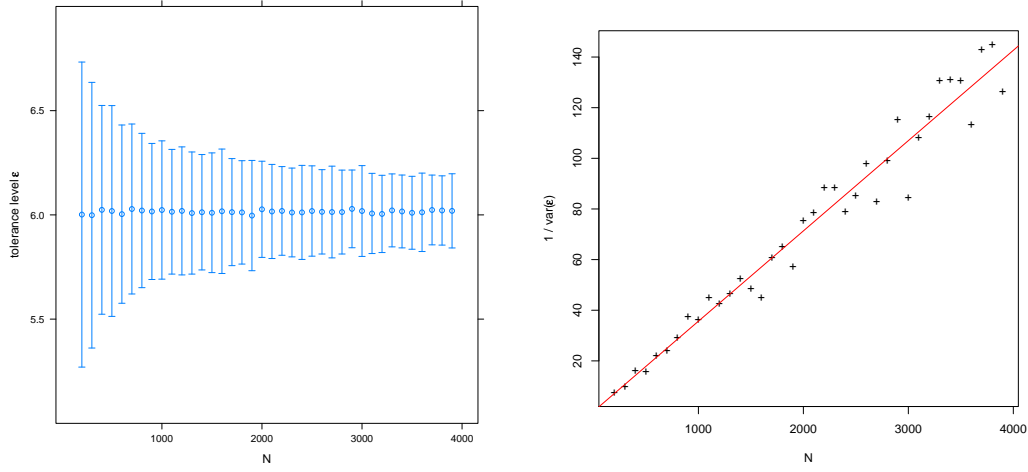


Figure 1: *Left*: confidence intervals of probability 0.95 for the calibrated tolerance level ε at the end of the first iterative stage when N increases. *Right*: inverse of ε 's variance with respect to N . Variances and confidence intervals were computed on 250 independent replications.

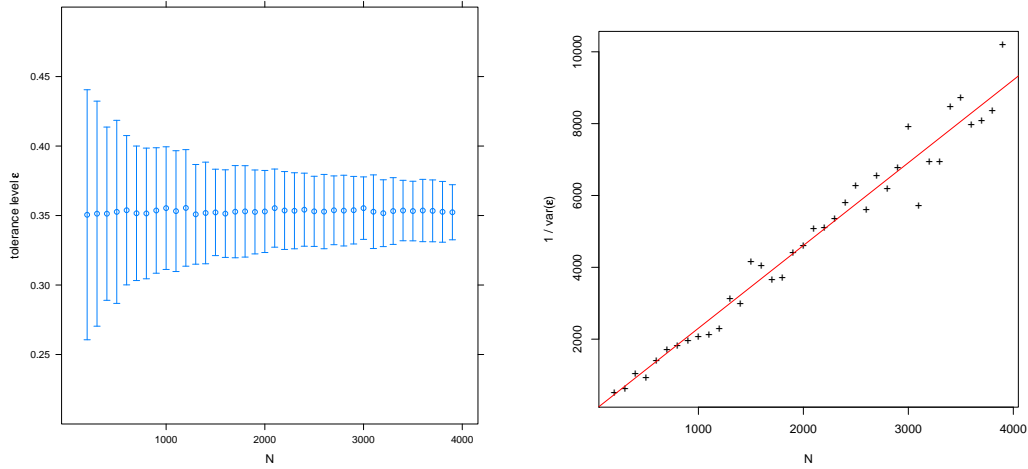


Figure 2: *Left*: confidence intervals of probability 0.95 for the calibrated tolerance level ε at the end of the 8th iterative stage when N increases. *Right*: inverse of ε 's variance with respect to N . Variances and confidence intervals were computed on 250 independent replications.

the four studies, we initialize our scheme with a particle system obtained by acceptance-rejection with a quantile level α_0 . Thus, we conclude that our ABC-SMC scheme reduces significantly the number of simulations according to the model when the prior provides little information (see the bottom curves in figure 3).

When the prior density support is very large, the sequential scheme needs many iterations to detect the relevant region for the particle simulation. This can be very inefficient and may have a strong negative impact on the gain factor. The choice of the quantile level α_0 is important in this situation. A good choice of α_0 avoids the region where learning is useless. The first ABC approximation $\pi_{\varepsilon_0}(\boldsymbol{\theta}|\mathbf{x}_{\text{obs}})$ assigns a very low mass to those parts of the prior density support. We illustrate this choice of α_0 in figure 3 (bottom curves). With prior density support $[-100, 100]$ and $\alpha_0 = 1/4$, the maximum of the gain factor is 14. This choice of α_0 is obtained using a system of particles drawn from the joint distribution $\pi(\boldsymbol{\theta})f(\boldsymbol{\theta}|\mathbf{x})$. The level α_0 is chosen such that the variance of $\boldsymbol{\theta}_i$ selected after one stage of acceptance-rejection is significantly reduced. More specifically, the choice $\alpha_0 = 1/4$ reduces the standard deviation of $\boldsymbol{\theta}$ by a factor 4.

4.2 A population genetics example

We now consider an experiment that relates more directly to the genesis of ABC, namely population genetics. We propose to perform a Bayesian inference of the parameters of an evolutionary scenario. This scenario retraces the invasion of bees in Europe. The scenario is presented in Figure 4. It consists of five inter-population events which occurred at instants t_1, \dots, t_5 (from the oldest to the newest). It is composed of two types of events: three divergences parameterized by their dates t_1, t_2, t_4 and two admixtures parameterized by their dates t_3, t_5 and rates r_1 and r_2 .

This scenario involves six populations that have diverged at different moments, as described in Figure 4. It includes two unobserved populations. We use different colors to distinguish the evolution of the different populations in the Figure 4.

The datasets for this study are composed of different samples of individuals collected from four populations observed until present and denoted by $Pop1, \dots, Pop4$ in Figure 4. Each sample is of the order of fifty. Our datasets consist of individual genotyping at eight independent microsatellite loci. To perform the ABC analysis, we consider 30 summary statistics selected from the statistics available in the *DIYABC* software of Cornuet et al. (2008).

The GSM model is considered in the study. The mutation rate is given per unit of time and per individual. For the i^{th} loci of the study, the total mutation $\mu_{i,\text{GLO}}$ rate is

divided into two parts as follows:

$$\mu_{i,\text{GLO}} = \mu_i + \mu_{i,\text{SNI}}.$$

μ_i is the rate that corresponds to an increase or a decrease of the DNA sequence by a length $G \times m$, where G is a realisation of a geometric random variable with parameter 0.42 and m is a pattern length. The rate $\mu_{i,\text{SNI}}$ is associated to the addition or removal of one nucleotide.

Priors.

Denote by Ne_1, \dots, Ne_6 the effective population sizes associated to the six populations involved in the scenario. This size varies with the population. Recall that inter-population events in the scenario occurred at t_1, \dots, t_6 . Thus, by regrouping the admixture rates r_1, r_2 and the mutation rates $\mu_i, \mu_{i,\text{SNI}}$ we obtain a vector of parameters of dimension 15.

The priors on the parameter vector are given as follows

- We suppose that

$$Ne_i \sim \text{Gamma}_{[0.1; 500000]}(\overline{Ne}/2, 2),$$

where $\text{Gamma}_{[a; b]}(p, s)$ is the gamma distribution with position and shape parameters p, s truncated to $[a; b]$. We have $\mathbb{E}[Ne_i | \overline{Ne}] = \overline{Ne}$, and \overline{Ne} is an hyper-parameter with uniform law $\mathcal{U}_{[5; 100,000]}$.

- the priors associated to the different event dates are given as $t_5 \sim \mathcal{U}_{[0.01; 800]}$, $t_4 - t_5 \sim \mathcal{U}_{[0; 50,000]}$, $t_3 - t_5 \sim \mathcal{U}_{[0; 50,000]}$, $t_2 - \max(t_3, t_4) \sim \mathcal{U}_{[100; 500,000]}$, $t_1 - t_2 \sim \mathcal{U}_{[100; 2000,000]}$.
- The priors on the mutations parameters are given as

$$\mu_i \sim_{\text{iid}} \text{Gamma}_{[5 \times 10^{-7}; 5 \times 10^{-2}]}(\bar{\mu}/2, 2),$$

where $\bar{\mu} = 10^{-4}$ and $\mathbb{E}[\mu_i] = \bar{\mu}$. We also have

$$\mu_{i,\text{SNI}} \sim_{\text{iid}} \text{Gamma}_{[10^{-9}; 10^{-3}]}(\bar{\mu}_{\text{SNI}}/2, 2),$$

where $\bar{\mu}_{\text{SNI}} = 10^{-6}$ and $\mathbb{E}[\mu_{i,\text{SNI}}] = \bar{\mu}_{\text{SNI}}$.

- The priors on admixture rates r_1 and r_2 are uniform on $[0.01; 0.999]$.

Figure 5 presents the results for the estimates of the parameters θ s defined for each population as $\theta = 4Ne\mu$. One can see that the replication power of the introduced

algorithm is significative. Figure 6 shows the posterior approximations for the different admixture rates. Finally, the gain factor for our proposal compared to the standard ABC acceptance-rejection algorithm is given in Figure 7. The dashed vertical line indicates the stopping time of the iterative algorithm, i.e., the first T for which $\rho_T \leq 0.1$. We stop the algorithm after 14 iterations. We only need half the number of simulations from the model to obtain results similar to the standard ABC acceptance-rejection algorithm.

5 Discussion

We proposed a self-calibrated ABC-SMC algorithm adapted from Del Moral et al. (2012) such that the number of simulations from the model is minimized. On a very challenging population genetics example, we shown that our proposal runs twice as fast as traditional ABC algorithms.

Our ABC-SMC scheme stops when the acceptance rate of Hastings-Metropolis falls below a fixed threshold. It seems possible to estimate the gain factor r_t and stop when it falls below 1. Typically, for each iteration, we must be able to estimate the probability that the distance to the observation is smaller than ε_t

$$\mathbb{P} \left(d(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon_t \right),$$

where the distribution of \mathbf{x} is the marginal of the joint distribution $(\boldsymbol{\theta}, \mathbf{x}) \sim \pi(\boldsymbol{\theta})\ell(\mathbf{x}|\boldsymbol{\theta})$. This is equivalent to estimating the number of simulations required for the ABC acceptance-rejection sampler to accept ESS_t particles. We have

$$\begin{aligned} \mathbb{P} \left(d(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon_t \right) &= \mathbb{P} \left(d(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon_t \middle| d(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon_{t-1} \right) \dots \\ &\quad \mathbb{P} \left(d(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon_1 \middle| d(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon_0 \right) \\ &\quad \mathbb{P} \left(d(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon_0 \right). \end{aligned}$$

Thus, we can envision to estimate the probability of the rare event

$$\left\{ d(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon_t \right\}$$

by the product of the levels of quantiles α_t calibrated by our scheme. At iteration t , \mathbf{x} is distributed according to the marginal of $\pi(\boldsymbol{\theta})f(\mathbf{x}|\boldsymbol{\theta})$ given $d(S(\mathbf{x}), S(\mathbf{x}_{\text{obs}})) \leq \varepsilon_{t-1}$. It would be very interesting to analyze the theoretical properties of this criterion.

References

- Beaumont, M. A. (2010). Approximate Bayesian Computation in Evolution and Ecology. *Annu. Rev. Ecol. Evol. Syst.*, 41:379–406.
- Beaumont, M. A., Cornuet, J.-M., Marin, J.-M., and Robert, C. P. (2009). Adaptive approximate Bayesian computation. *Biometrika*, 96(4):983–990.
- Cappé, O., Guillin, A., Marin, J. M., and Robert, C. P. (2004). Population Monte Carlo. *J. Comput. Graph. Statist.*, 13(4):907–929.
- Cornuet, J.-M., Santos, F., Beaumont, M., Robert, C. P., Marin, J.-M., Balding, D., Guillemaud, T., and Estoub, A. (2008). Inferring population history with DIY ABC: a user-friendly approach Approximate Bayesian Computation. *Bioinformatics*, 24(23):2713–2719.
- Csilléry, K., Blum, M., Gaggiotti, O., and François, O. (2010). Approximate Bayesian Computation (ABC) in practice. *Trends in Ecology and Evolution*, 25(7):410–418.
- Del Moral, P. (2004). *Feynman-Kac formulae*. Probability and its Applications (New York). Springer-Verlag, New York. Genealogical and interacting particle systems with applications.
- Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential Monte Carlo samplers. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 68(3):411–436.
- Del Moral, P., Doucet, A., and Jasra, A. (2011). On adaptative resampling procedures for sequential Monte Carlo methods. *Bernoulli*, 18(1):252–278.
- Del Moral, P., Doucet, A., and Jasra, A. (2012). An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Stat. Comput.*, page (to appear).
- Douc, R. and Moulines, E. (2008). Limit theorems for weighted samples with applications to sequential Monte Carlo methods. *Ann. Statist.*, 36(5):2344–2376.
- Doucet, A., de Freitas, N., and Gordon, N., editors (2001). *Sequential Monte Carlo methods in practice*. Statistics for Engineering and Information Science. Springer-Verlag, New York.
- Drovandi, C. C. and Pettitt, A. N. (2011). Estimation of parameters for macroparasite population evolution using approximate Bayesian computation. *Biometrics*, 67(1):225–233.
- Liu, J. S. (2008). *Monte Carlo strategies in scientific computing*. Springer Series in Statistics. Springer, New York.

- Marin, J.-M., Pudlo, P., , Robert, C. P., and Ryder, R. (2012). Approximate Bayesian computational methods. *Stat. Comput.*, page (to appear).
- Marjoram, P., Molitor, J., Plagnol, V., and Tavaré, S. (2003). Markov chain Monte Carlo without likelihoods. *Proc. Natl. Acad. Sci. USA*, 100(26):15324–15328.
- Pritchard, J., Seielstad, M., Perez-Lezaun, A., and Feldman, M. (1999). Population growth of human Y chromosomes: a study of Y chromosome microsatellites. *Molecular Biology and Evolution*, 16(12):1791–1798.
- Sisson, S., Fan, Y., and Tanaka, M. (2009). Sequential Monte Carlo without likelihoods: Errata. *Proc. Natl. Acad. Sci. USA*, 106(39):16889.
- Sisson, S. A., Fan, Y., and Tanaka, M. M. (2007). Sequential Monte Carlo without likelihoods. *Proc. Natl. Acad. Sci. USA*, 104(6):1760–1765 (electronic).

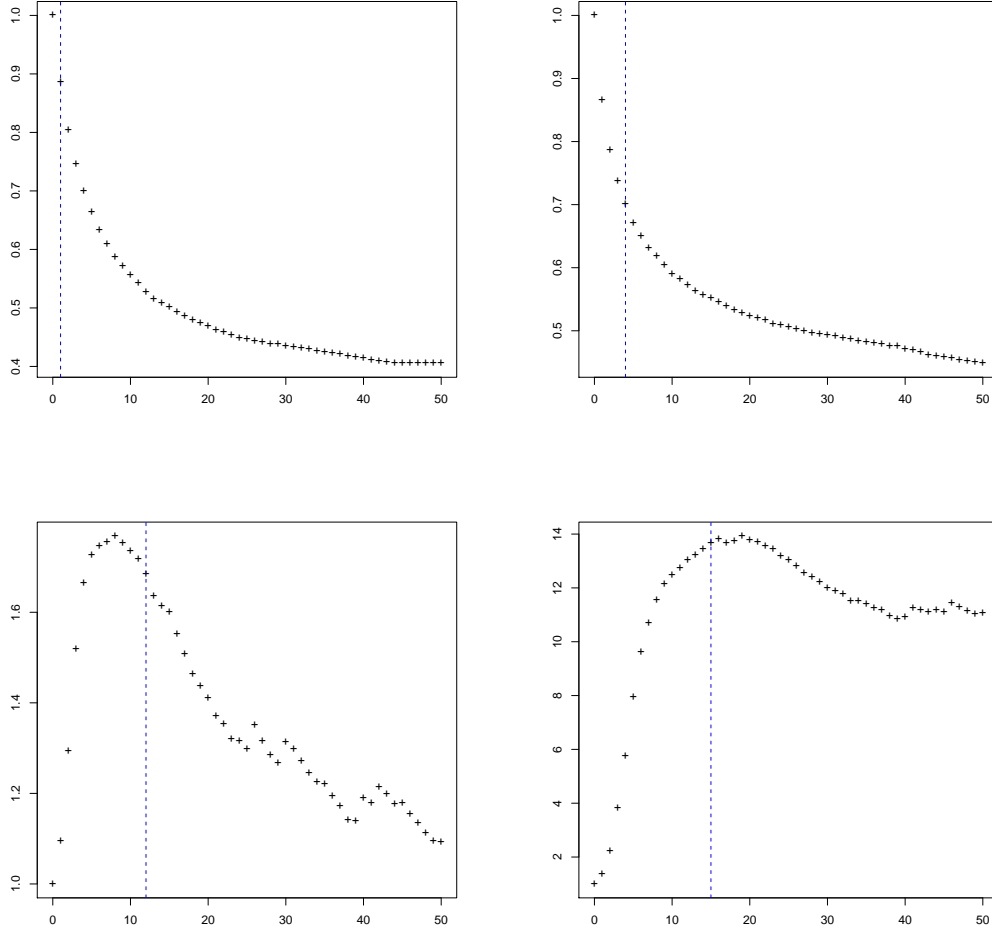


Figure 3: Study of the efficiency of our proposal based on the information provided by the prior. *Top, left:* the prior density support is $[-0.1; 0.1]$ and $\alpha_0 = 1/10$. *Top, right:* the prior density support is $[-1; 1]$ and $\alpha_0 = 1/10$. *Bottom, left:* the prior density support is $[-10; 10]$ and $\alpha_0 = 1/2$. *Bottom, right:* the prior density support is $[-100; 100]$ and $\alpha_0 = 1/4$.

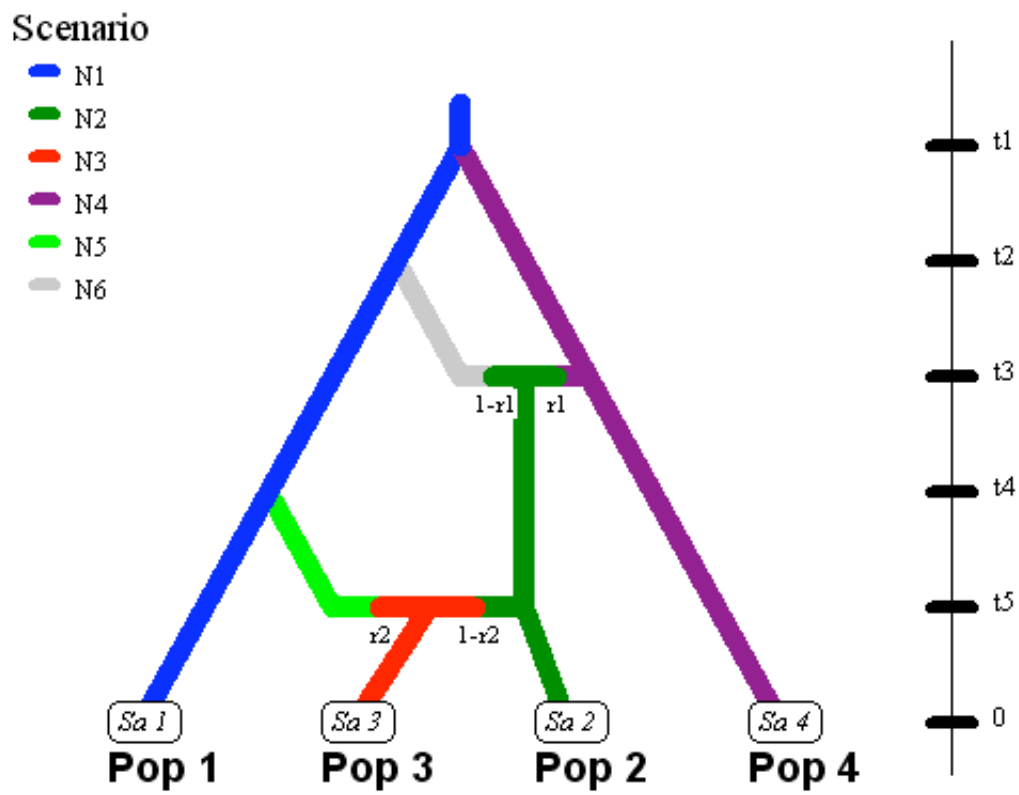


Figure 4: Population scenario for the honeybees dataset

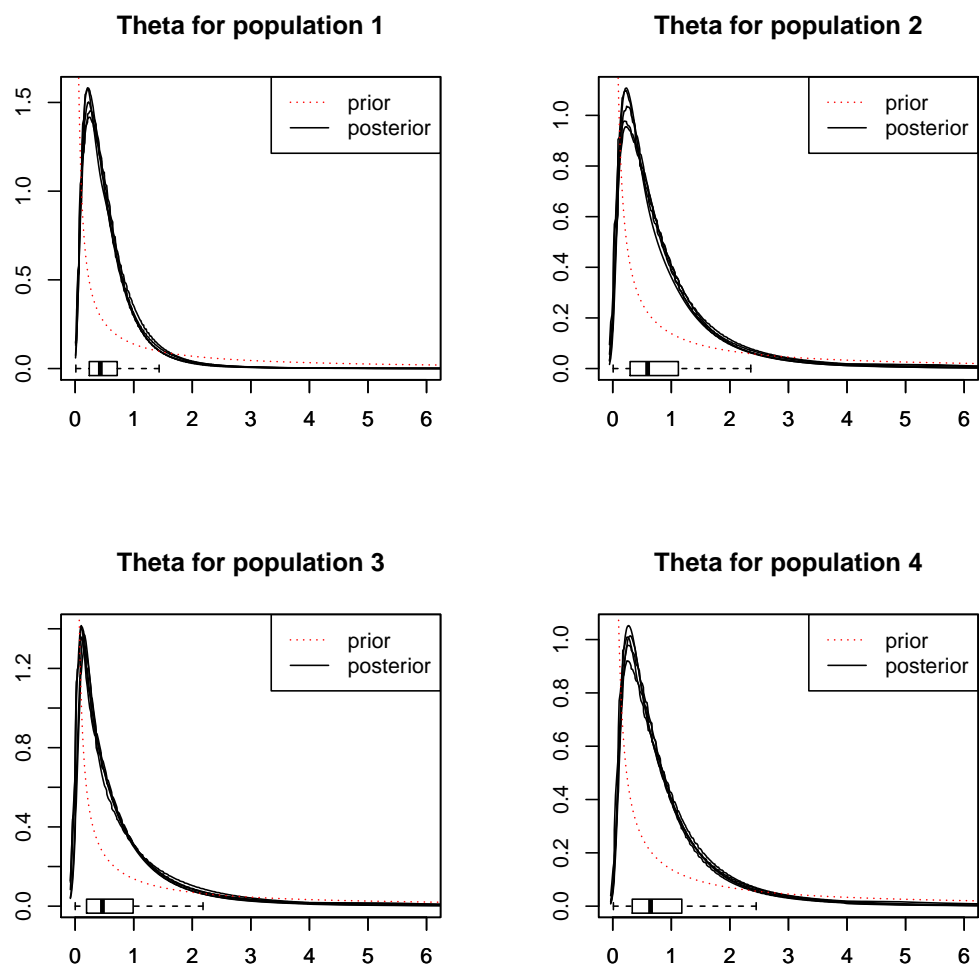


Figure 5: Estimates of the posterior distributions of the θ_i 's for five independent replicates

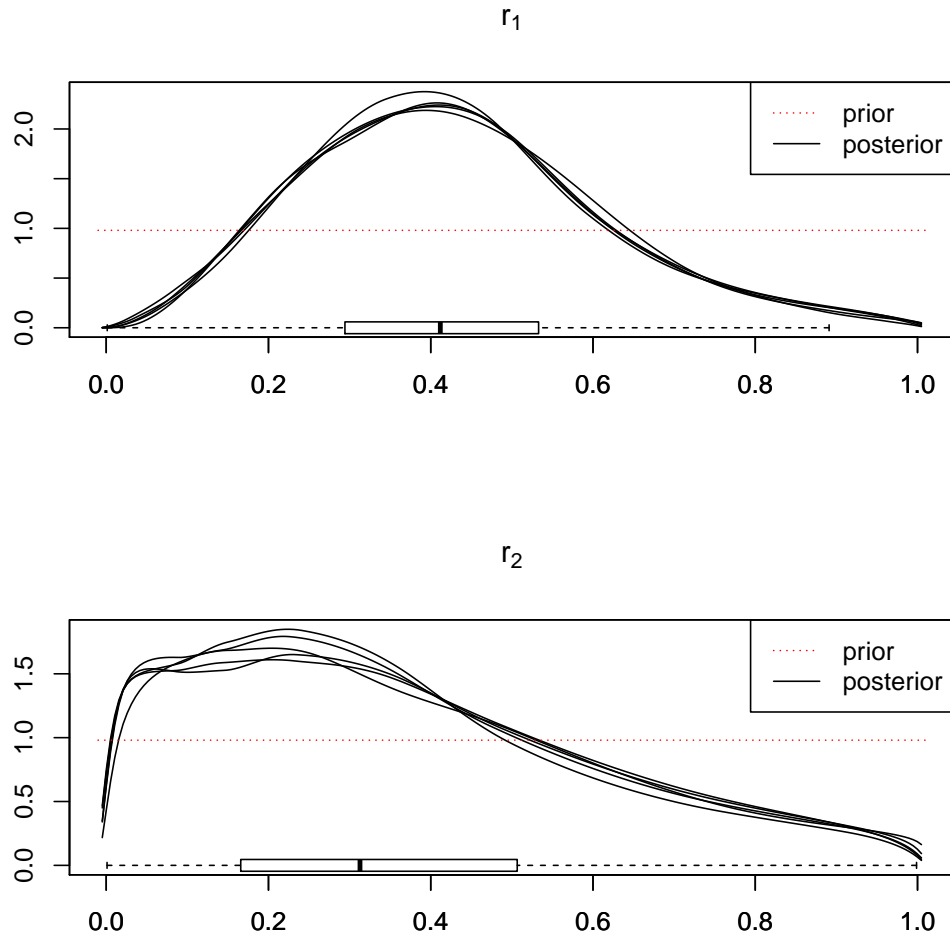


Figure 6: Estimates of the posterior distribution of r_1 and r_2 for five independent replicates

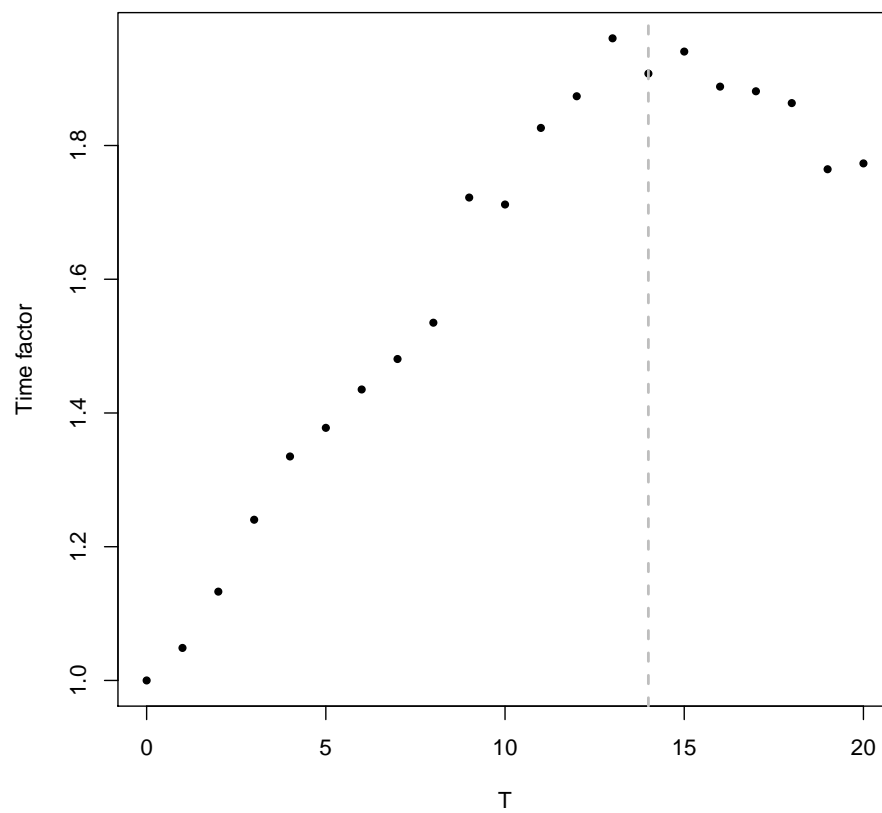


Figure 7: Time factor for the population genetics example